**2. PASOS PREVIOS A LA PROGRAMACIÓN DE ORDENADORES.**

**ALGORIT­MIA.**

Ahora que ya conocemos el funcionamiento del ordenador nos toca saber cómo hacerlo funcionar. Para ello vamos a explicar unas nociones básicas.

Como todos sabemos que realizar una película no es un trabajo fácil, primero se debe escribir el guión, después buscar los actores que mejor se adapten a los distintos papeles, encontrar un buen director, rodar la película, repetir escenas que no han salido como quería el director, etc. También, para realizar un programa de ordenador lo primero que se debe hacer es un guión (qué datos serán los de entrada, como los procesará, cuales serán los datos de salida, etc), **el guión de un programa se conoce con el nombre de algoritmo**. Posteriormente se elige el lenguaje de programación que mejor se adapte al algoritmo, buscar un buen programador, escribir el programa, repetir partes que no hacen lo que quería el programador, etc.

La algoritmia es una herramienta común a todos los lenguajes de programación, por ello los programadores prefieren aprender un algoritmo que un programa. Así, si les dan el algoritmo del programa lo entenderán, mientras que el programa no.

¿Se imagina usted a un director de cine que haga primero la película y después llame a expertos para que le digan si la película es comercial, si el argumento está desfasado, si hay muchas películas parecidas en el mercado, etc?. Lo lógico es que el director cuando tiene el guión llame al equipo de colaboradores, hagan los cambios oportunos y se ruede la película. En el primer caso el director se arriesga a que la película no sea aceptada por el público y tendrá que rodarla de nuevo adaptándose a las exigencias del público; en el segundo caso la película saldrá al mercado con garantías de éxito. Lo mismo ocurre cuando se escribe un programa, primero se realiza el algoritmo y sobre él se estudia si el programa es muy grande, si realizará las operaciones que estaban previstas, incluso con el algoritmo se puede ver si el programa funcionará correctamente. **Si deseamos realizar un buen programa deberemos previamente haber realizado un buen algoritmo**.

Los algoritmos se realizan con un conjunto de símbolos que iremos describiendo en la wiki, para cada sentencia de programación suele haber un símbolo equivalente en algoritmia, no obstante existen una serie de condiciones que debe cumplir cualquier algoritmo:

* **Finitud**. Todo algoritmo debe finalizar en algún momento, no podemos hacer un algoritmo que no acabe nunca. Además cuantos menos pasos tenga el algoritmo mejor será éste. El proceso siguiente es un ejemplo de una combinación de pasos que nunca finalizaría:

1. Hacer I=1

2. Escribe el valor de I.

3. Incrementar el valor de I en una unidad.

4. Ir al paso 2.

Además todo algoritmo debe tener un punto de inicio (primer paso realizado) y un punto final (último paso realizado).

* **Definibilidad.** Cada paso se debe entender perfectamente y estar definido. Por ejemplo:
* Escribir datos personales**.**

Es un paso que se entiende pero no está definido, ya que no sabemos cuáles son los datos personales a los que se refiere, en su lugar se debería escribir:

* Escribir nombre, dirección, teléfono**.**
* **Entrada**. La entrada suele ser información que o bien se suministra desde fuera del algoritmo, o bien se expresa dentro del algoritmo con unas condiciones iniciales. Las entradas están representadas por un paso y una información. El paso está representado en el algoritmo y dice qué acción se lleva a cabo para realizar la entrada; la información especi­fica qué tipo de dato se utilizará en la entrada.

1. Leer Nombre del usuario del fichero de datos.

2. Esperar a que el usuario introduzca por el teclado su edad.

En el primer paso se indica que el nombre está almace­nado en algún lugar y se ha de ir a buscarlo. El tipo de datos es carácter. En el segundo paso se indica que el usuario debe introducir por teclado su edad y el ordenador debe esperar a que el usuario realice esa acción. El tipo de datos es numérico.